

ABSTRACT OF THE DISCLOSURE

An asynchronous programming language that facilitates concurrent programming by utilizing futures, asynchronous calls, and joins on futures. For each of a client interface and a service interface of an asynchronous interface, respective models are automatically extracted. A behavioral contract is defined on the asynchronous interface. The client and service models are then passed to modular checking algorithm that checks to ensure that both the client and the service conform to the behavioral contract. The checking algorithm combines region-based type systems with model checking techniques to handle pointer aliasing in a sound manner.